

CLAIMS

What is claimed is:

1. A method comprising:

receiving a request identifying an application;

5 determining whether the application corresponding to the request is present within a designated search path;

generating the application based upon automatic compilation of one or more source files, if it is determined that the application is not present within the designated search path;

10 determining whether regeneration of the application is needed if it is determined that the application is present within the designated search path; and

regenerating the application based upon automatic compilation of the one or more source files, if it is determined that regeneration is needed.

15 2. The method of claim 1, wherein the one or more source files correspond to one or more prioritized source file types specified in an ordered set of compilation rules.

3. The method of claim 1, wherein generating the application comprises:

identifying a target file corresponding to the application;

20 accessing an ordered set of compilation rules associated with the identified target file, each of the ordered set of compilation rules identifying a target file type and a corresponding source file type for use in generating the associated target file;

selecting a compilation rule from the ordered set of compilation rules;

determining whether a source file corresponding to the source file type of the selected compilation rule exists within a designated source path for building the identified target file;

building the identified target file based upon the selected compilation rule if it is

5 determined that the source file exists within the designated source path; and.

updating a build date/time associated with the application.

4. The method of 3, wherein if a source file corresponding to the source file type of the selected compilation rule exists within the designated source path, a representation
10 of the source file is stored in a first persistent data structure, and

wherein if a source file corresponding to the source file type of the selected compilation rule does not exist within the designated source path, a derived source file representation is stored in a second persistent data structure.

15 5. The method of claim 4, wherein the representation of the source file comprises a file name and path associated with the corresponding source file.

6. The method of claim 4, wherein determining whether regeneration of the application is needed comprises determining if any source files corresponding to
20 members of said first persistent data structure have a date/time more recent than the build date/time of the application.

7. The method of claim 6, wherein regenerating the application comprises automatically compiling source files corresponding to members of said first persistent data structure that have a date/time that is more recent than the build date/time of the application .

5

8. The method of claim 4, wherein determining whether regeneration of the application is needed comprises determining if any members of said second persistent data structure are present within said designated source path.

9. The method of claim 4, wherein at least one of the first and second persistent data structures comprises a list.

10. The method of claim 3, wherein the source path and the search path are equivalent.

11. The method of claim 1, wherein the request is received through a network connection as one or more data packets.

12. A method comprising:

identifying a target file corresponding to an application identified by a request;
accessing an ordered set of compilation rules associated with the identified target file, each of the ordered set of compilation rules identifying a target file type and a corresponding source file type for use in generating the associated target file;

selecting a compilation rule from the ordered set of compilation rules; and
determining whether a source file corresponding to the source file type of the
selected compilation rule exists within a designated source path for building the
identified target file,

5 wherein if a source file corresponding to the source file type of the
selected compilation rule exists within a designated source path, a representation
of the source file is stored in a first persistent data structure, and

 wherein if a source file corresponding to the source file type of the
selected compilation rule does not exist within the designated source path, a
derived source file representation is stored in a second persistent data structure.

13. The method of claim 12, further comprising:

 receiving a request identifying an application, wherein said application
corresponds to said identified target file;

5 determining whether the application is present within a designated search path;
and

 automatically compiling the application based at least in part upon the contents of
said first and second persistent data structures and in accordance with said ordered set
of compilation rules, if it is determined that the application is not present within the
20 designated search path.

14. The method of claim 13, wherein the source path and the search path are
equivalent.

15. The method of claim 13, further comprising:

determining if regeneration of the application is needed if it is determined that the application is present within the designated search path;

5 regenerating the application if it is determined that regeneration is needed; and
updating a build date/time associated with the application.

16. The method of claim 15, wherein determining whether regeneration of the application is needed comprises determining if any source files corresponding to members of said first persistent data structure have a date/time more recent than the build date/time of the application.

17. The method of claim 15, wherein determining whether regeneration of the application is needed comprises determining if any members of said second persistent data structure are present within said designated source path.

18. The method of 12, wherein at least one of said first and second persistent data structures comprise a list.

20 19. An article of manufacture comprising:

a storage medium having stored therein a plurality of programming instructions,
which when executed operate to

receive a request identifying an application;

determine whether the application corresponding to the request is present within a designated search path;

facilitate generation of the application based upon automatic compilation of one or more source files, if it is determined that the application is not present within the designated search path;

determine whether regeneration of the application is needed if it is determined that the application is present within the designated search path; and

facilitate regeneration of the application based upon automatic compilation of the one or more source files, if it is determined that regeneration is needed.

20. The article of claim 19, wherein the one or more source files correspond to one or more prioritized source file types specified in an ordered set of compilation rules.

21. The article of claim 19, wherein the instructions to facilitate generation the application comprise instructions to

identify a target file corresponding to the application;

access an ordered set of compilation rules associated with the identified target file, each of the ordered set of compilation rules identifying a target file type and a corresponding source file type for use in generating the associated target file;

select a compilation rule from the ordered set of compilation rules;

determine whether a source file corresponding to the source file type of the selected compilation rule exists within a designated source path for building the identified target file;

facilitate building of the identified target file based upon the selected compilation rule if it is determined that the source file exists within the designated source path; and update a build date/time associated with the application.

5 22. The article of 21, wherein if a source file corresponding to the source file type of the selected compilation rule exists within the designated source path, a representation of the source file is stored in a first persistent data structure, and

wherein if a source file corresponding to the source file type of the selected compilation rule does not exist within the designated source path, a derived source file representation is stored in a second persistent data structure.

23. The article of claim 22, wherein the representation of the source file comprises a file name and path associated with the corresponding source file.

15 24. The article of claim 22, wherein the instructions to determine whether regeneration of the application is needed comprise instructions to determine if any source files corresponding to members of said first persistent data structure have a date/time more recent than the build date/time of the application.

20 25. The article of claim 24, wherein the instructions to regenerate the application comprise instructions to facilitate automatic compilation of source files corresponding to members of said first persistent data structure that have a date/time that is more recent than the build date/time of the application.

26. The article of claim 22, wherein determining whether regeneration of the application is needed comprises determining if any members of said second persistent data structure are present within said designated source path.

5

27. The article of claim 22, wherein at least one of the first and second persistent data structures comprises a list.

28. The article of claim 21, wherein the source path and the search path are equivalent.

29. The article of claim 19, wherein the request is received through a network connection as one or more data packets.

30. An article of manufacture comprising:
a storage medium having stored therein a plurality of programming instructions,
which when executed operate to

identify a target file corresponding to an application identified by a request;

access an ordered set of compilation rules associated with the identified

target file, each of the ordered set of compilation rules identifying a target file type and a corresponding source file type for use in generating the associated target file;

select a compilation rule from the ordered set of compilation rules; and

determine whether a source file corresponding to the source file type of the selected compilation rule exists within a designated source path for building the identified target file,

wherein if a source file corresponding to the source file type of the selected compilation rule exists within a designated source path, a representation of the source file is stored in a first persistent data structure, and

wherein if a source file corresponding to the source file type of the selected compilation rule does not exist within the designated source path, a derived source file representation is stored in a second persistent data structure.

31. The article of claim 30, wherein the instructions further comprise instructions to receive a request identifying an application, wherein said application corresponds to said identified target file;

determine whether the application is present within a designated search path;

facilitate automatic compilation of the application based at least in part upon the contents of said first and second persistent data structures and in accordance with said ordered set of compilation rules, if it is determined that the application is not present within the designated search path; and

update a build date/time associated with the application.

32. The article of claim 31, wherein the source path and the search path are equivalent.

33. The article of claim 31, wherein the instructions further comprise instructions to
5 determine if regeneration of the application is needed if it is determined that the application is present within the designated search path; and
facilitate regeneration of the application if it is determined that regeneration is needed.

34. The article of claim 33, wherein the instructions to determine whether
10 regeneration of the application is needed further comprise instructions to determine if any source files corresponding to members of said first persistent data structure have a date/time more recent than the build date/time of the application.

35. The article of claim 33, wherein the instructions to determine whether
15 regeneration of the application is needed further comprise instructions to determine if any members of said second persistent data structure are present within said designated source path.

20 36. The article of 30, wherein at least one of said first and second persistent data structures comprise a list.